Fast semantic segmentation of aerial images based on color and texture

Mohaddeseh Ghiasi
Department of Electrical and Computer
Isfahan University of Technology
Isfahan, Iran
m.ghiasifathabad@ec.iut.ac.ir

Rassoul Amirfattahi
Department of Electrical and Computer
Isfahan University of Technology
Isfahan, Iran
fattahi@cc.iut.ac.ir

***Abstract*— In this paper, a semantic segmentation method for aerial images is presented. Semantic segmentation allows the task of segmentation and classification to be performed simultaneously in a single efficient step. This algorithm relies on descriptors of color and texture. In the training phase, we first manually extract homogenous areas and label each area semantically. Then color and texture descriptors for each area in the training image are computed. The pool of descriptors and their semantic label are used to build two separate classifiers for color and texture. We tested our algorithm by KNN classifier. To segment a new image, we over-segment it into a number of superpixels. Then we compute texture and color descriptors for each superpixel and classify it based on the trained classifier. This labels the superpixels semantically. Labeling all superpixels provides a segmentation map. We used local binary pattern histogram fourier features and color histograms of RGB images as texture and color descriptors respectively. This algorithm is applied to a large set of aerial images and is proved to have above 95% success rate.**

Keywords*— Semantic Recognition, Texture descriptors, aerial images, superpixels.

I. INTRODUCTION

Automatic segmentation and recognition of aerial images is a requirement for autonomous landing systems to decide the safest possible area for landing an autonomous aircraft. There has been various attempts in the literature to design real-time and accurate methods of segmentation that also try to overcome different challenges posed to analysing images taken from cameras mounted on airplanes and satellites. Such images are often lack the detailed texture areas that can sufficiently identify the texture type. Shadows and reflections pose a major challenge to process such images especially in water areas where reflection can completely alter the texture. Varying viewpoints and rotation angles are a natural characteristic of such images. The number of different segments are often much higher than many segmentation methods can deal with. This problem arose from the fact that these images are taken from high altitude. In addition, large changes in altitude pose a significant challenge to those segmentation algorithm which rely on feature descriptors such as SIFT [1]. Finally time requirement and accuracy demands make the entire process even more complicated.

Our motivation for using a semantic segmentation framework arises from the fact that for an autonomous aircraft, a mere segmentation is not sufficient. We also require to an understanding of the input image to assess the possibility of a safe landing. This means a recognition task has to be immediately followed after segmentation. Combing these two stages in a single efficient can be easily accomplished by applying a semantic segmentation method.

There have been several attempts for segmentation of aerial images. Gupta *et al.* [2] have used maximum likelihood classification for the segmentation of aerial images. They combined it with a certainty based fusion criterion. This algorithm is part of a more complex system, which is currently being designed to assist an operator in updating an old map of an area using aerial images. An advantage of their algorithm is combining colour and texture. However, they used autoregressive models for texture description, which are not rotation and scale invariant. Cao *et al.* [3] employed a level set algorithm for segmentation of aerial images. They used a 3-level wavelet decomposition of a small area of the image as rotation invariant features for a pixel. They perform the segmentation task in pixel-level which is computationally expensive. Wei *et al.* [4] have proposed a multiphase level set evolution scheme for aerial image segmentation using multi-scale image geometric analysis.

This paper presents a simple, fast and efficient framework for semantic segmentation of aerial images. Semantic segmentation combines the two important and often tedious tasks of segmentation and recognition in a single step.

our algorithm relies on descriptors of color and texture. The underlying reason of incorporating color information to assist texture clues arises from the fact that, in aerial imagery, some textures might look similar despite belonging to different categories. This is because shadows, reflections, low resolution and weather conditions introduce some undesirable artifacts to texture descriptors. We compensate for these artifacts by incorporating color information.

In the training phase, we first manually extract homogenous areas and label each area semantically. Then color and texture descriptors for each area in the training image are computed. The pool of descriptors and their semantic label are used to build a classifier. The training phase is described in the training section.

To segment a new image, we over-segment it into a number of superpixels. Superpixels provide an automatic uniform and homogenous splitting of the input image. Using superpixels has the important advantage of reducing computational cost as segmentation is performed in superpixel level rather than pixel level. Then we compute texture/color descriptors for each superpixel and apply KNN to classify each superpixel. This labels the superpixels semantically. Labeling all superpixels provides a segmentation map.

We employed Local Binary Pattern Histogram Fourier features (LBP-HF) [5] to describe texture. This decision is based on a thorough evaluation study on different texture descriptors as stated in this paper. LBP-HF is globally rotation invariant and has been proved to outperform many state-of-the-art texture descriptors. Color histograms of RGB images are incorporated as color descriptors.

As we will show in the Experimental Results section, this algorithm is applied to a large set of aerial images and is proved to have above 95% success rate (average). As boundaries of superpixels coincide with the image edges, the algorithm is capable of preserving boundaries. Occluded areas are successfully segmented and labeled similar. Since the segmentation is performed in super--pixel level (not the pixel level), the training phase is very fast and effective. This is a very important advantage of our algorithm. Finally as experiments show, shadows and areas of brightness reflection have been segmented correctly.

II. The Proposed Semantic Segmentation Algorithm

In this section, the proposed segmentation algorithm is explained. First in the Training section, we discuss the training procedure and then the test phase of the proposed algorthm is described thoroughly.

A. Training

We first manually extract some homogenous rectangular areas from images of the training dataset and label each area semantically.

The size of these areas is not important and they can be of varying sizes. This is because the texture and colour descriptors should be invariant to the texture size and we normalize those descriptors (histograms).

To describe texture of an area we apply LBP-HF to the area. This descriptor is fully described in the following section. To describe colour of an area, we compute colour histograms of RGB channels and combine them in a single histograms. The histogram bins are of size 32. We then build our classifier based on both colour and texture descriptors. In our experiments, we tested our algorithm by KNN classifier.

B. Splitting image into superpixels

Here we describe different steps by which we segment an input image based on the KNN trees build in the training step. This segmentation algorithm segments an image in superpixel-level as opposed to pixel-level. We first split the input image to a number of superpixels which are homogenously textured areas produced by an over segmentation. many methods such as local variation [6], mean-shift [7], Ncuts [8] and watershed [9], can be used to generate superpixels. One advantage of super-pixels is that they accurately preserve object boundaries. Furthermore, subsequent stages benefit from the reduced image grid which in turn reduces computational complexity. We used Turbopixels proposed by Levinshtein *et al.* [10]. This method generates more uniformly sized superpixels and is observed to be much faster. This decision is based on our qualitative evaluation study of different methods to produce superpixels. We generate 500 superpixels for each input image. Fig. 1 shows the superpixels obtained by the TurboPixels algorithm for two aerial images.

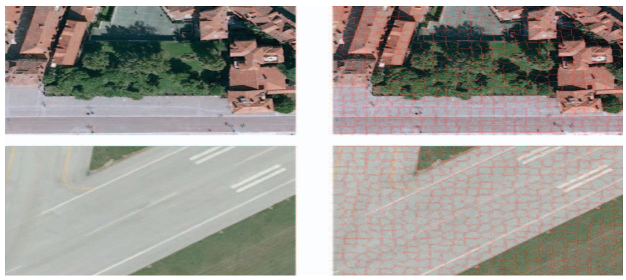

Fig. 1. Original image (left), suprpixels (right)

C. Computing LBP-HF descriptors

Local Binary Pattern (LBP) to describe texture was introduced by Ojala *et al.* [11] and since then this descriptor has undergone many improvements. To compute LBP descriptors for a textured area, first a LBP code is computed for each pixel. To compute the LBP code for each pixel, a circular symmetric neighborhood of radius R about it is assumed. Then P sample points on this circular neighborhood are considered to contribute to the LBP code of the central pixel. Each sample point is threshold at the value of the central pixel. If it has a greater or equal value to the central pixel, it will be assigned a "1" bit and otherwise a "0" bit. The P bits are arranged to obtain a binary code (based on a binary weighting of the circular neighborhood). This operator is denoted as $LBP_{P,R}$. A textured area could be described by the normalized histogram of the binary codes computed for every pixel inside the area. One way to make LBP operator rotation invariant is to repeatedly apply a circular bit-wise right shift to the LBP code of each pixel till the minimum value is reached. This minimum value is considered to a rotation invariant LBP code for that pixel. As a result, LBP histograms of the textured area remain the same. This process makes the LBP code rotation invariant globally and locally. This is because the process by which LBP becomes rotation invariant is performed in the pixel-level. However, in many applications, we do not desire LBP descriptors to be invariant locally. This is because it ignores the local directions of texture. To solve this problem Matas *et al.* [5] proposed keeping the LBP codes as they are (without making them rotation invariant) and compute the LBP histograms. If there is global rotation in the texture, there will be a circular shift in the histogram. Assume one computes the Fourier descriptors on these histograms. A circular shift in the input of Fourier transform, results in a phase shift in the output but the Fourier magnitude will not change. This means that the magnitude of Fourier descriptor computed on LBP histograms can be used as a rotation invariant texture descriptor.

It should be noted that as Ojala *et al.* [11] reported some particular LBP codes generally have a much higher frequency in real-world textures. It is reasonable to give a higher importance to those LBP codes. They suggest that those LBP codes that have either one or two transitions from "0" to "1" or vice versa construct a significant number of LBP codes seen in real-world textures (above 90%). They refer to these LBP codes as "uniform" and to the rest as "non-uniform" LBP patterns. They assign a single LBP code to all non-uniform patterns. This follows that the LBP code for a uniform pattern can be the number of ones in the pattern, i.e. LBP codes for uniform patterns range from 0 to P+1. The code of P+2 is assigned to all non-uniform patterns. This method of encoding for LBP patterns is called "uniform LBP". Matas *et al.* [5] also rely on uniform LBP in LBP-HF.

In our final experiments, we evaluated different versions of LBP. They include LBP-HF, multiresolution LBP (with $LBP_{8,1}$ and $LBP_{16,2}$). We also tested a more recent version of LBP, evaluated LBPV [12] which takes into account contrast information. However, as LBP-HF had a higher performance the descriptor of choice for our experiment is LBP-HF.

D. *Computing color histograms*

We have experimented with a number of different color spaces including RGB and HSV color spaces. However, RGB color spaced was shown to be the most effective one.

E. *Applying KNN classifier*

We built two KNN classifiers based on LBP-HF descriptors and RGB histograms separately. In the test phase, to semantically label each superpixel we search these trees. Both KNN trees vote for each superpixel. We accept the decision made by the KNN tree that generates the higher vote. If they vote for two different classes equally then we choose the decision made by color unless the exceptional case where texture has chosen "water" class and color has chosen "grass" class. For KNN classifier this procedure is illustrated in Fig. 2.

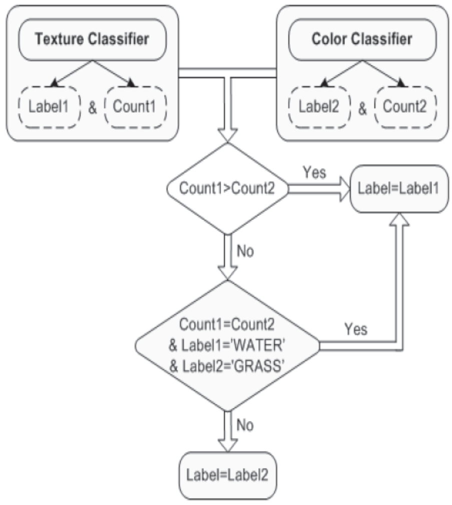

Fig. 2. The procedure to decide for the label of a superpixel. The variable count1 is the number of votes achieved by the texture KNN tree and count 2 is the number of votes achieved by the color KNN tree.

This heuristic is based on the error analysis shown in TABLE I. Repeating this step for all superpixels generates a label map which is a segmentation of the original image.

TABLE I. A COMPARISON BETWEEN THE ERRORS PRODUCED BY THE KNN TREES MADE BASED ON RGB HISTOGRAMS AND LBP-HF. THIS SHOWS THAT FOR THE WATER CLASS, THE LBP-HF IS A MORE RELIABLE CLUE.

	LBP-HF	**RGB**
building	16.4%	3.6%
Road	7.3%	0%
Grass	3.6%	3.6%
Tree	0%	3.6%
Water	7.3%	12.7%
total	6.9%	4.7%

An important consideration when using a KNN tree is the distance measure. For the KNN tree made by LBP-HF we tested histogram intersection, Chi square, L2 norm and L1 norm metrics. Our experiments show that L2 norm is the preferred metric.

III. EXPERIMENTAL RESULTS

To evaluate the proposed approach, we applied it to a set of aerial images downloaded from Google Earth. These images are taken from the altitude of 150 meters from city of Venice. The image size is of 700×1024 pixels. We are interested in the following 5 semantic classes: Building, Road, Grass, Tree and Water.

Fig. 3 show some of the segmentation results generated by the algorithm. We do not apply any post-processing step to the results as it was not required.

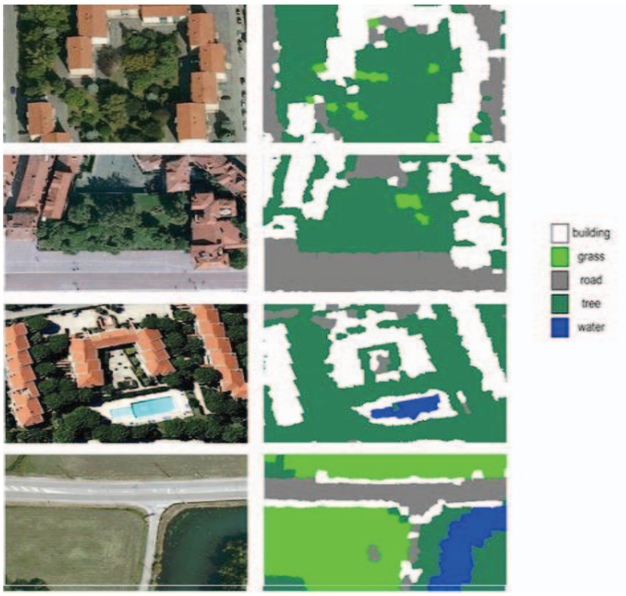

Fig. 3. Original image (left), segmentation map-KNN (right),

To provide a quantitative metric of the performance of the proposed method, we have asked a user to manually segment 10 randomly chosen images. Then for each pixel we have compared the label produced by the algorithm to the label assigned by the user (the ground truth image). We then computed the confusion matrix as shown in TABLE II. In the worst case (for road semantic class) we have achieved 92% success rate.

TABLE II. A COMPARISON BETWEEN THE ERRORS PRODUCED BY THE KNN TREES MADE BASED ON RGB HISTOGRAMS AND LBP-HF. THIS SHOWS THAT FOR THE WATER CLASS, THE LBP-HF IS A MORE RELIABLE CLUE.

	KNN				
	Building	**Road**	**Grass**	**Tree**	**Water**
Building	**96.67**	0.23	0	3.1	0
Road	2.88	**92.98**	0.19	3.7	0.25
Grass	0.92	0.18	**95.71**	3.17	0.02
Tree	1.01	0.42	0.59	**97.85**	0.13
Water	0.13	0.05	0.21	1.85	**97.76**

IV. CONCLUSION

In this paper, we have presented a method for semantic aerial image segmentation which has been applied to a large, real dataset of aerial images. We segmented an image according to the following 5 semantic classes: Building, Road, Grass, Tree and Water. Our algorithm allows simultaneous segmentation and classification and fuses color and texture to make a robust decision for labeling each area. We used LBP-HF and color histograms of RGB images as texture and color descriptors respectively. We trained two separate classifiers for color and texture. To test an input image, it is over-segmented into a number of superpixels. Then we compute texture or color descriptors for each superpixel and let the classifiers to decide for the label of the superpixel separately. Labeling all superpixels provides a segmentation map. This algorithm is applied to a large set of aerial images and is proved to have above 95% success rate.

REFERENCES

[1] D. G. Lowe, “Distinctive image features from scale-invariant keypoints”, *International journal of computer vision*, Vol. 60, No. 2, pp. 91-110, 2004.

[2] M. P. Dubuisson-Jolly and A. Gupta, "Color and texture fusion: application to aerial image segmentation and GIS updating," *Image and Vision Computing,* vol. 18, pp. 823-832, 2000.

[3] G. Cao, Z. Mao, X. Yang, and D. Xia, "Optical aerial image partitioning using level sets based on modified Chan-Vese model," *Pattern Recognition Letters,* vol. 29, pp. 457-464, 2008.

[4] W. Wei, Y. Xin, and C. Guo, "A Multiphase Level Set Evolution Scheme for Aerial Image Segmentation Using Multi-scale Image Geometric Analysis," *Structural, Syntactic, and Statistical Pattern Recognition,* pp. 56-64, 2006.

[5] T. Ahonen, J. Matas, C. He, and M. Pietikainen, "Rotation invariant image description with local binary pattern histogram fourier features," *Image Analysis,* pp. 61-70, 2009.

[6] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *International Journal of Computer Vision,* vol. 59, pp. 167-181, 2004.

[7] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Transactions on pattern analysis and machine intelligence,* pp. 603-619, 2002.

[8] S. X. Yu and J. Shi, "Multiclass spectral clustering,", pp. 313-319 vol.311,2003.

[9] L. Vincent and P. Soille, "Watersheds in digital spaces: an efficient algorithm based on immersion simulations," *IEEE transactions on pattern analysis and machine intelligence,* pp. 583-598, 1991.

[10] A. Levinshtein, A. Stere, K. N. Kutulakos, D. J. Fleet, S. J. Dickinson, and K. Siddiqi, "Turbopixels: Fast superpixels using geometric flows," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* pp. 2290-2297, 2009.

[11] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on pattern analysis and machine intelligence,* pp. 971-987, 2002.

[12] Z. Guo, L. Zhang, and D. Zhang, "Rotation invariant texture classification using LBP variance (LBPV) with global matching," Pattern Recognition, vol. 43, pp. 706-719,2010.